An integrated order batching and picker routing problem

Martin Bué[1], Diego Cattaruzza[2], Maxime Ogier[2], Frédéric Semet[2]

[1]: Inria Lille-Nord Europe
martin.bue@inria.fr

[1]: Univ. Lille, CNRS, Centrale Lille, Inria
UMR 9189 - CRIStAL
Centre de Recherche en Informatique Signal et Automatique de Lille
F-59000 Lille, France
diego.cattaruzza,maxime.ogier,frederic.semet@centralelille.fr

1 Introduction

In this paper, we focus on the warehouse management problem faced by our industrial partner HappyChic[1]. HappyChic is a French firm specialized in men's ready-to-wear. HappyChic gathers three leading brands: Jules, Brice, and Bizzbee. The logistic department of HappyChic is in charge of the three warehouses (one for each brand), and all the related logistics activities. In this paper, we consider operations in the warehouse dedicated to clothing products of the brand Jules, located in the North of France, close to the city of Lille. The warehouse supplies the shops of Jules in all France and some shops abroad. Thus, the usual customers to satisfy are the shops to which new clothing products are shipped.

The warehouse under consideration is depicted in Figure 1 and consists in four main zones: (1) the *reception zone* where products delivered by suppliers are received; (2) the *storage zone* where cartons are stored in racks; (3) the *picking zone* where pickers pick items from cartons and prepare boxes to be sent to customers; (4) the *delivery zone* where trucks are loaded with boxes shipped to customers.

All along the article, we make the distinction between *carton* and *box*. A carton contains items of a single product. Cartons are received by the warehouse on pallets and stored in the storage zone. They are then moved to the picking zone. There, items are picked from the cartons and put into boxes by pickers. Thus, a box contains customer demands and possibly items of different products.

Trucks carrying products from the suppliers arrive at the *reception zone* of the warehouse. Here dedicated employees unload the trucks, and pallets of cartons are prepared to be stored. Then, employees driving forklifts move cartons from the reception zone into the racks of the *storage zone*. Other employees are dedicated to supplying the *picking zone*. They pick up cartons from pallets in the storage zone and bring them, using forklifts, into the picking zone. They are allowed only to go through cross-aisles 1, 3 and 5 and to leave cartons at the top or the bottom of the different aisles (see solid blue lines in Figure 1).

The picking zone is composed of 4 blocks (see Figure 1). Let us call these blocks, block 1 to 4. Each block is equipped with racks where cartons with products are located. Contrarily to standard cases presented in the literature (see for example Scholz and Wäscher (2017)), we distinguish the two sides of the racks: the front and the back side. Cartons are inserted into racks from the back side and products are picked from the front side. Racks are situated in such a way that a front (resp. back) side always faces another front (resp. back) side of another rack (except, of course, for the first and the last rack). This placement defines a front-aisle and a back-aisle.

Back-aisles are used by employees to bring cartons from the top or the bottom of an aisle to their exact location (see dotted blue lines in Figure 1). Front-aisles are used by employees called *pickers* to collect products from cartons.

The cross-aisles in the picking zone are not all walkable by pickers. For security reasons, cross-aisles 1, 3 and 5 are dedicated to forklifts that supply the picking zone from the storage zone. Pickers

[1] http://www.happychicgroup.com

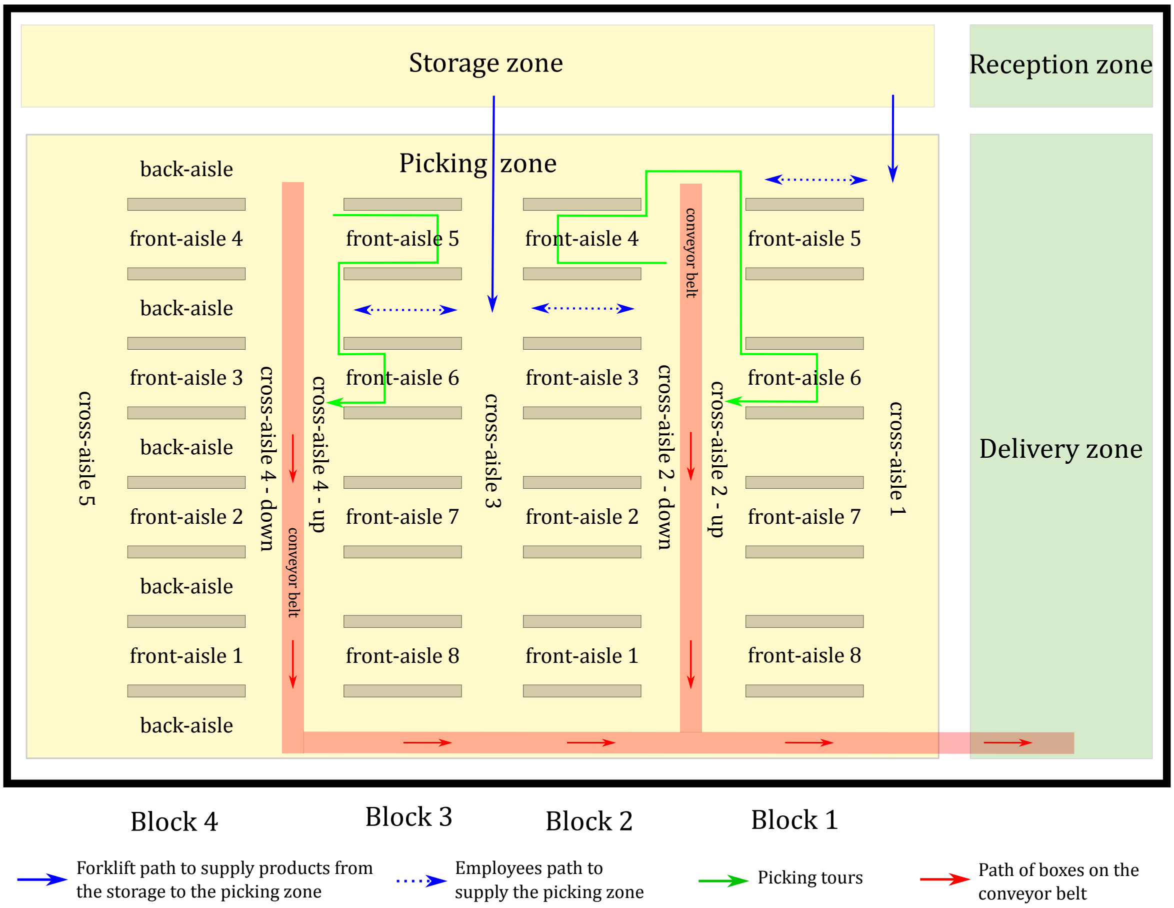


Figure 1: The warehouse.

are allowed to walk only across cross-aisles 2 and 4. Moreover, these cross-aisles are equipped with a conveyor belt that automatically brings boxes prepared by pickers from the picking zone to the *delivery zone*. The conveyor belt divides cross-aisles 2 and 4 into two sub-cross-aisles. Let us use the words *up* and *down* to distinguish between these two sub-cross-aisles.

Pickers walk around the warehouse pushing a trolley that contains up to f boxes. Due to the particular warehouse layout, the pickers cannot cross the cross-aisle 3. As a consequence, picking tours are limited to blocks 1 and 2 or to blocks 3 and 4. Moreover, all the pickers follow the same circular path orientation and always walk following a clockwise walkway. This is due to space limitations in the aisles, and also facilitates picking operations since all routes are in the same direction. Thus, in a given route, the locations names follow a topological order. We call this particular policy the *walking policy*. It is formally defined in Section 3. In this paper, we focus on the optimization of the operations in the picking zone, namely the work of the pickers.

Let us illustrate the picking tours with an example. We consider blocks 1 and 2 and number the aisles as follows. Front-aisle and back-aisle numbered 1 are the bottommost of block 2. Other aisles are numbered clockwise. Enumeration of blocks 3 and 4 follows the same rule. Only enumeration of the front-aisles is reported in Figure 1 due to space limitation. A possible picking tour (solid green lines in Figure 1) starts at front aisle 5 and continues to front aisles 6 and 7. The picker then puts the boxes prepared on the conveyor belt and starts another picking tour. The tour may start from the front aisle 7, or the picker may walk back through cross-aisle up 2 and cross-aisle down 2 and starts a new picking tour.

Each working day a set of demands has to be prepared. The demands are associated with customers, i.e. shops located throughout France and abroad. Based on the sales, the shop demands are computed during the night. Demands usually consist of several items which may not be put into a single box. The demand of each customer has then to be split in different boxes. Clearly one box must contain only products associated with a single customer.

The problem under study consists in optimizing the productivity in the picking zone by simultaneously determining: (1) the products that constitute each box to be delivered to customers; and (2) the batch of boxes to constitute a picking tour.

2 Literature Review

Warehouse management involves different optimization problems such as conceiving picking tours, batching orders, storage assignment, layout design, and zoning (see, for example, de Koster et al. (2007)). Decisions to optimize productivity in a warehouse can have different horizons and be classified as strategic, tactical and operational. Strategic decisions determine the layout of the warehouse, the disposition of each zone (receiving, storage, picking and delivery) with respect to the others, the decisions that involve the storage policy as well as picking policy (automated versus human). Tactical decisions can involve the location of products based on their forecast demand in the storage and picking zone. At the operational level, storage and picking tours need to be efficiently determined and coordinated. Storage tours move products from the storage to the picking zone, while picking tours aim at collecting items to satisfy the demand of customers (de Koster et al. (2007), Marchet et al. (2015)).

Picking operations can be classified based on the automatism introduced in the process. As in Marchet et al. (2015) five classes can be identified: picker-to-parts, parts-to-picker, pick-to-box, pick-and-sort, and completely automated picking. In the first category, pickers move around the warehouse to pick items while in the parts-to-picker class, automated devices bring loads to pickers that are in charge of picking the right quantity required by the order under consideration. In pick-to-box systems, pickers are assigned to different zones and boxes containing customer orders move using a conveyor through different zones to be filled. In pick-to-sort systems, orders are batched and pickers collect items of a certain product to satisfy orders in the whole batch. A conveyor brings all the items to the sorting area where items are assigned to orders. Finally, humans do not intervene in full automated picking systems.

Picking systems where the human presence is necessary for operations still involve the majority of the warehouses (de Koster et al. (2007)). Recent studies (*Ecommerce Europe* (2017), Marchet et al. (2015)) show how the cost of the automation would be too high to be profitable in a short or mid-term horizons. For these reasons and since in the problem under study picking operations can be classified as picker-to-parts, we concentrate the review on papers related to this area.

In the HappyChic case, our interest is focused on the operational decisions for picking operations. It concerns order batching and picker routing problems. Usually, order batching is performed when the size of orders is smaller than the capacity of the trolley used to collect the items. Order batching consists in grouping a set of orders into picking tours to minimize the total distance traveled by pickers. This problem is NP-hard (Gademann and van de Velde (2005)). Given a set of locations to visit, the picker routing problem aims to decide the sequence of locations to visit to minimize the traveled distance. These two problems have been studied extensively as separate problems (de Koster et al. (2007)). Some recent works have addressed the joint order batching and picker routing problem. Valle et al. (2017) propose a branch-and-cut algorithm based on a mixed integer programming formulation, and Scholz and Wäscher (2017) proposed an iterated local search approach.

In the HappyChic case, we face a specific batching problem. Indeed, the size of an order is usually larger than the capacity of a trolley. Hence, the batching problem has two levels. First, items of a given order have to be batched into boxes. Second, boxes have to be assigned to trolleys. It is noteworthy that these two batching levels are mutually dependent.

3 Problem definition

In this section, we formally introduce the problem and the notation used in the paper. All along the following sections, we suppose that each working day is divided into T periods. We indicate by $\mathcal{T} = \{1, \ldots, T\}$ the period set. The warehouse considered contains a set of different products $\mathcal{P}$. With

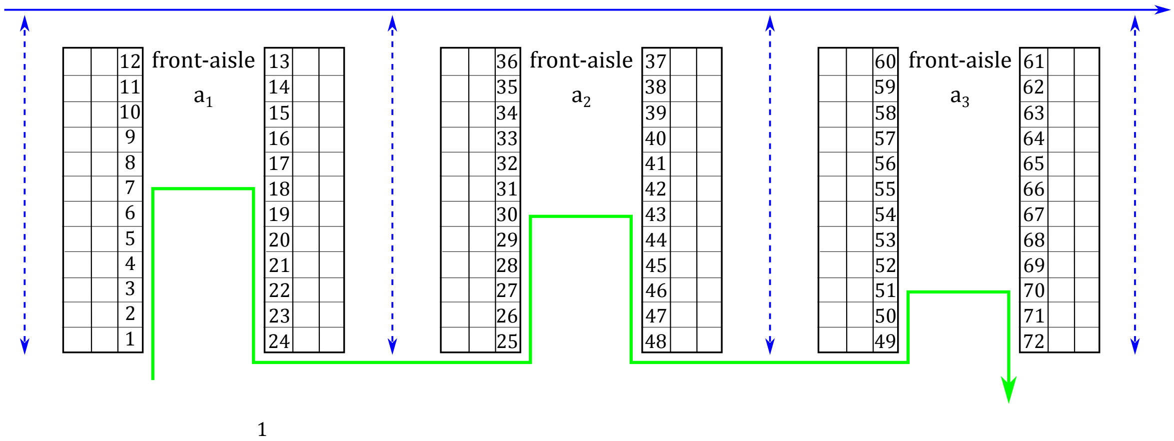


Figure 2: Front-aisles and back-aisles. Legend as in Figure 1. Locations are numbered following the walking order.

each product $p \in \mathcal{P}$ is associated the volume v_p and the weight w_p of a single item.

3.1 Picking zone

The picking zone of the warehouse consists of four blocks each containing different aisles having racks on one or both sides. Racks are divided in several locations that contain cartons. Locations are filled from the back and emptied from the front. This defines what we call *front-aisles* and *back-aisles* (see Figure 2). Only front-aisles are involved in the picking operations.

An order is imposed on the front-aisles of the picking zone. First, aisles of blocks 1 and 2 and aisles of blocks 3 and 4 are ordered starting from the bottommost aisle of blocks 2 and 4 to the bottommost aisle of blocks 1 and 3 following the clockwise sense. Let $\mathcal{A}$ be the set of front-aisles. Given two front-aisles $a_1, a_2 \in \mathcal{A}$, aisle a_1 precedes aisle a_2 and we write $a_1 \prec a_2$, if a_1 comes before a_2 in this given order.

The set of all possible locations in the aisles of the picking zone is indicated by $\mathcal{L}$. A particular order is imposed on locations. Let us define the function $a : \mathcal{L} \to \mathcal{A}$ that associates each location with its aisle. Given two locations l_1 and l_2 such that $a(l_1) \neq a(l_2)$, we say that l_1 precedes l_2, and we indicate it by $l_1 \prec l_2$, if $a(l_1) \prec a(l_2)$. On the other hand, if $a(l_1) = a(l_2)$, we define the following rule. Entering a front-aisle from the cross-aisle, locations on the left are ordered from the beginning to the end of the front-aisle, while locations on the right are ordered on the opposite way. Thus, when $a(l_1) = a(l_2)$, we write $l_1 \prec l_2$ if:

- l_1 is on the left-side of the aisle, while l_2 on the right;
- l_1 and l_2 are both on the left-side and l_1 is nearer to the beginning of the aisle than l_2;
- l_1 and l_2 are both on the right-side and l_1 is nearer to the end of the aisle than l_2.

This order is called *walking order* or *walking policy*. An example is given in Figure 2, where the numbers on each location follow the walking order. Given two locations l_1 and l_2, distances d_{l_1,l_2} and d_{l_2,l_1} are determined. They represent the distance of the path that goes from one location to the other following the walking order. Due to logistic constraints in the warehouse, it can be impossible to reach location l_2 from location l_1 (and/or vice versa). In this case, the distance is infinite. Location 0 represents a location where a tour starts and ends. Picking tours start and end at any place along the conveyor belt. This can be approximated by the entry point or exit point of an aisle. Hence location 0 is a fictive point that represents entrance or exit of an aisle. We then consider distance $d_{0,l}$ as the distance from the entrance point of aisle $a(l)$ to location l, and distance $d_{l,0}$ as the distance from location l to the exit point of aisle $a(l)$.

At a given period $t \in \mathcal{T}$, each location contains items of the same product. However, different locations in the picking zone may contain the same product. For each period t, we assume that the storage level in the picking zone is sufficient to satisfy the total demand.

3.2 Pickers

A team of employees is in charge of collecting items from the cartons in the picking zone and to fill the boxes that will be delivered to shops. These employees are called *pickers*. It is supposed that the number of pickers is large enough to ensure the preparation of the total demand to be prepared within a given period.

A picker walks around the warehouse respecting the walking policy. The picker pushes a trolley with up to f boxes available. He/she picks products from cartons and put them into the boxes. Once the all required items are picked, he/she walks to the conveyor belt on which he/she places the boxes. He/she then starts another picking tour with a trolley containing empty boxes. Then, the picker is allowed to walk against the walking policy to locate him/herself at the beginning of the next tour. Note that, due to the particular walking policy considered in the warehouse, the set of locations to visit to pick the items directly implies the path that a picker needs to follow to collect the items.

3.3 Demands

A set of demands $\mathcal{D}$ is to be prepared during a working day. The set $\mathcal{D}$ is partitioned in T subsets/periods, in which the working day is divided. Subset $\mathcal{D}_t$, $t \in \mathcal{T}$, contains all the demands that need to be prepared during period t.

For each demand $d \in \mathcal{D}$, a set of products $\mathcal{P}_d = \{p_1^d, \dots, p_{n_d}^d\} \subseteq \mathcal{P}$ needs to be collected. A quantity q_j^d and a location l_j^d are associated with each product p_j^d in demand d. This corresponds to items of the same product, situated at the same location, and needed to prepare d. Note that different items of the same product could be picked from cartons located in different locations in the picking zone. We assume that given product p_j^d in demand d, its location l_j^d is set. This assumption is satisfied for the HappyChic case since they use a FIFO rule to empty the different locations associated with the same product.

Whenever useful for the presentation of this work, a set notation is used as follows. Given a demand d, $p \in d$, $p \in \mathcal{P}$ means that product p is required by demand d. Moreover, if an index is not necessary, it is removed to ease reading. For instance, q_d^p indicates the quantity of product p required in demand d.

3.4 Boxes

Boxes to be delivered to shops must respect a maximal volume v_{box}^{max} and a maximal weight w_{box}. Moreover, each box must satisfy a minimal volume v_{box}^{min} of products. A box, that is not filled at least as much as the minimal volume, can be crushed by other boxes during transportation. Filling up a box increases its strength. Note that it may happen that a demand is too small to ensure the minimal volume capacity constraint. It is then the decision of the managers of the warehouse to decide if this demand is prepared (with violation of the capacity constraint) or delayed to another day. Here, we minimize the number of boxes that violated the minimal volume constraint and consider that they are integrated into the proposed picking tours.

3.5 Objective and constraints

The problem consists in simultaneously determining for each period $t \in \mathcal{T}$: (1) the partition of each demand $d \in \mathcal{D}_t$ into boxes; (2) the batching of boxes into groups of up to f boxes, i.e., the determination of a picking tour in order to minimize the number of boxes that violate the minimal volume constraint and the distance covered by the pickers while satisfying the maximal volume and weight constraints, the walking policy imposed into the picking zone, and the trolley capacity.

Note that the objectives are prioritized. First, we minimize the number of boxes that violate the minimal volume constraint, and second, the walking distance. Moreover, even if the objective is to minimize the total distance walked by the pickers, the industrial partner wants to ensure the number of tours remains reasonable. Indeed, since there is not a unique starting location (a tour can start at any place along the conveyor belt), minimizing the distance could lead to generate a large set of small tours. This is not acceptable since starting a new tour requires some time for the picker, that can be viewed as a fixed cost. Hence, we impose a fixed distance penalty on all tours.

4 Algorithmic approach

In this section, we describe the algorithm that has been developed to tackle the problem. Our algorithm includes two phases. The first phase splits each demand into different boxes (Section 4.1) while the second phase generates picking tours once the boxes have been determined (Section 4.2).

4.1 Splitting demands

Given a particular demand d, it may happen that the items cannot be contained into a single box or it may be advantageous to consider several boxes to increase the pickers' productivity. In these cases, the demand needs to be divided into several boxes able to contain it.

To perform this task, we apply the following algorithm inspired from the procedure *split* proposed by Beasley (1983) and successfully used by Prins (2004) in the context of the vehicle routing problem. This procedure is based on the computation of shortest paths on acyclic graphs. The algorithm we propose, seeks for a distribution of a given demand in several boxes such that volume and weight constraints are respected while increasing productivity, i.e. yield to operational plans with the shortest possible walking distance.

Let us suppose that products $\{p_1, p_2, \ldots, p_n\}$ are required to satisfy demand d. Let us associate a quantity q_i with each product p_i included in demand d. It follows that demand d requires a total of $I = \sum_{i=1}^{n} q_i$ items.

Let us assume that each item $i \in \mathcal{I}$, $\mathcal{I} = \{1, \ldots, I = \sum_{i=1}^{n} q_i\}$ composing demand d has to be collected from the location l_i of the picking zone and let us suppose, without loss of generality, that items are ordered following the walking policy, namely such that $i, j \in \mathcal{I}$, $i < j$ implies $l_i \prec l_j$ (see definition in Section 3.1).

The total volume of demand d is $v_d = \sum_{i=1}^{n} q_i v_{p_i}$, and the total weight of demand d is $w_d = \sum_{i=1}^{n} q_i w_{p_i}$. If $v_d < v_{box}^{min}$ and $w_d \leq w_{box}^{max}$, all items of demand d are put into a single box that violates the minimal volume constraint. In the following, we only consider demand d such that $v_d \geq v_{box}^{min}$ or $w_d > w_{box}^{max}$. It means that the items of such demands may be split into several boxes.

We construct a graph $\mathcal{G} = (\bar{\mathcal{I}}, \mathcal{E})$ where $\bar{\mathcal{I}}$ is the node set while $\mathcal{E}$ is the arc set. Set $\bar{\mathcal{I}}$ contains $1 + I$ nodes, numbered $0, 1, \ldots, I = \sum_{i=1}^{n} q_i$. Node 0 is a dummy node, while node $i \in \bar{\mathcal{I}}$, $i > 0$ represents the i-th item in $\mathcal{I}$ forming demand d.

With each node $i \in \bar{\mathcal{I}}$, we associate the quantities v_i (resp. w_i) the volume (resp. the weight) of the item represented by node i.

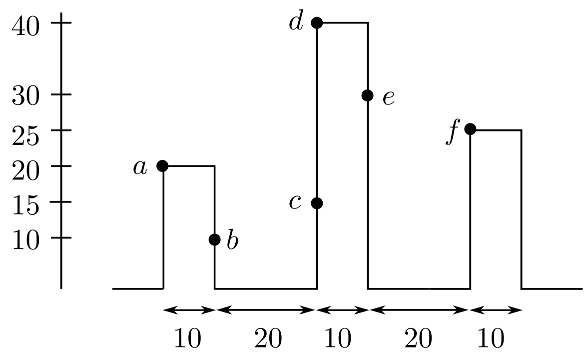
Then, for each couple of nodes $i, j \in \bar{\mathcal{I}}$ we calculate three quantities:

- $d_{ij} = d_{0,l_{i+1}} + \sum_{h=i+2}^{j} d_{l_{h-1},l_h} + d_{l_j,0}$, i.e., the walking distance needed to pick items $i+1, \ldots, j$ from their locations $l_{i+1}, \ldots, l_j$;
- $w_{ij} = \sum_{h=i+1}^{j} w_h$, i.e., the weight of items $i+1, \ldots, j$;
- $v_{ij} = \sum_{h=i+1}^{j} v_h$, i.e., the volume of items $i+1, \ldots, j$.

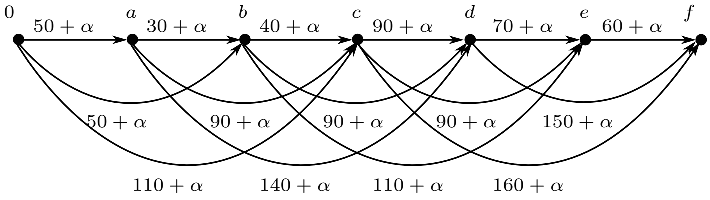
Based on these quantities, we can determine the arcs of graph $\mathcal{G}$ in such a way that each arc represents a feasible box. We first determine the arc set $\mathcal{E}_1$ defined as:

$$\mathcal{E}_1 = \{(i,j) | i < j,\ i,j \in \bar{\mathcal{I}}, v_{box}^{min} \leq v_{ij} \leq v_{box}^{max}, w_{ij} \leq w_{box}\}. \tag{1}$$

Second, in order to guarantee the graph connectivity, we consider the arc set $\mathcal{E}_2$ made of all arcs $(i-1, i)$, $i \in \bar{\mathcal{I}} \setminus \{0\}$ even if the associated box does not respect the constraint on the minimal volume. Thus,



(a) Data for the example.



(b) Graph for split procedure.

Figure 3: Example of the split procedure.

$$\mathcal{E}_2 = \{(i-1, i) | i \in \bar{\mathcal{I}} \setminus \{0\}, v_{i-1,i} < v_{box}^{min}\}. \tag{2}$$

We finally define the arc set $\mathcal{E} = \mathcal{E}_1 \cup \mathcal{E}_2$.

The cost c_{ij} associated with each arc $(i, j) \in \mathcal{E}_1$ is given by the quantity $\alpha^{BOX} + d_{ij}$, where α^{BOX} is a parameter that penalizes the creation of new boxes. This parameter α^{BOX} permits to limit the number of boxes, hence to propose a reasonable number of tours. The cost c_{ij} associated with each arc $(i, j) \in \mathcal{E}_2$ is given by $\delta^{BOX} + d_{ij}$, where δ^{BOX} is a large-enough value.

To determine the distribution of the demand d into boxes, we compute the shortest path from the dummy source node 0 to node I with respect to costs c_{ij} . Each arc belonging to the shortest path represents a box. The shortest path provides the best split of products into boxes given the order of products and the cost structure.

The value of δ^{BOX} should be such that an arc in $\mathcal{E}_2$ is selected only if needed to guarantee the existence of a path between nodes 0 and I. Whenever the shortest path contains an arc in $\mathcal{E}_2$ the corresponding box does not respect all the constraints.

The algorithm is repeated for each demand $d \in \mathcal{D}$ to obtain its distribution into boxes. All the boxes that are created are inserted in set $\mathcal{B}$.

An example of the split procedure is given in Figures 3a and 3b. Here the demand is composed of six different products noted a, b, c, d, e, f. They are located in the warehouse as illustrated in Figure 3a. Values on the ordinate axis represent the distance from the beginning of the aisle to the product location, while values on the abscissa represent intra-aisle and inter-aisle distances. Following this disposition, we have that $a \prec b \prec c \prec d \prec e \prec f$. We suppose that only one item per product is required. For simplicity, let us suppose that all the products have a volume and a weight equal to 10 units, the minimal volume for a box equals 10, while the maximal weight and volume equal 30 units.

The graph that is built with respect to this demand is reported in Figure 3b. A box that contains only product a is associated with a distance of $50 + \alpha^{BOX}$ units. The picker needs to get into the aisle and walk for 20 units to reach product a. He/she then needs to cross and to get out from the aisle for an additional distance of 30 units. This is represented by arc $(0, a)$. A box that contains products b and c is associated with a distance of $90 + \alpha^{BOX}$, that is the distance to get into the first aisle, walk until the position in front of b, cross the aisle to reach b, get out from the aisle, and walk to the next aisle for a total of 50 units. Then, the picker needs to get into the second aisle and walk until c. He/she then crosses the aisle and gets out of it for an additional 40 units of distance. Thus the total distance to compose the box is $90 + \alpha^{BOX}$. This is represented by arc (a, c).

Note that if $\alpha^{BOX} \leq 50$ the shortest path that goes from node 0 to node f is made by arcs $(0, b)$, (b, e) and (e, f) for a total cost of $220+3\alpha^{BOX}$. The demand is then split in three boxes. One containing products a and b, another with products c, d and e, the last with only product f. On the other case, when $\alpha^{BOX} > 50$, the best solution is to create only 2 boxes. The shortest path in this case is made by arcs $(0, c)$ and (c, f) for a total cost of $270 + 2\alpha^{BOX}$. The solution in this case considers two boxes, one with products a, b and c, the other with products d, e and f.

4.2 Batching boxes for picker trips

A picker pushes a trolley containing up to f boxes. We thus need to batch the boxes in order to obtain the trips that pickers have to perform. Let us suppose that a set $\mathcal{B} = \{1, \ldots B\}$ of boxes have to be prepared. Let a_b^S be the starting aisle associated with box $b \in \mathcal{B}$, namely the aisle where is located the first item to pick to prepare box b (with respect to the walking policy). Let a_b^T be the ending aisle associated with box $b \in \mathcal{B}$: the aisle where is located the last item to pick to prepare box b. Moreover, let d_b be the walking distance to pick all items of box b. We suppose, without loss of generality, that boxes are numbered such that $b_1 < b_2$, $b_1, b_2 \in \mathcal{B}$ if and only if

$$\begin{aligned} a_{b_1}^S \neq a_{b_2}^S \text{ and } a_{b_1}^S \prec a_{b_2}^S \quad &\text{or} \quad a_{b_1}^S = a_{b_2}^S \text{ and } a_{b_1}^T \prec a_{b_2}^T \\ &\text{or} \quad a_{b_1}^S = a_{b_2}^S \text{ and } a_{b_1}^T = a_{b_2}^T \text{ and } d_{b_1} > d_{b_2}. \end{aligned}$$

To create batches of boxes of cardinality up to f, we adapt the *split* procedure described in Section 4.1. Similarly, a graph $\mathcal{G} = (\bar{\mathcal{B}}, \mathcal{E})$ is constructed. The node set $\bar{\mathcal{B}}$ contains $1 + B$ nodes, numbered from 0 to B. Node 0 is a dummy node, while node $i = 1, \ldots, B$ represents the i-th box in $\mathcal{B}$. The arc set $\mathcal{E}$ contains all the possible batches of boxes, namely, all batches with up to f boxes. Thus

$$\mathcal{E} = \{(i,j) | i, j \in \bar{\mathcal{B}}, j - i \leq f\}$$

A cost c_{ij} is associated with each arc $(i,j) \in \mathcal{E}$ and represents the total walking distance needed to collect all items in boxes $i+1, i+2, \ldots, j$, plus a fixed penalty α^{TOUR} to start a tour. Note that c_{ij} can be computed keeping in memory all the aisles that contain items in the boxes under consideration and the location of the furthest item in each aisle.

The shortest path that goes from the dummy node 0 to node B provide a partition of the box set $\mathcal{B}$ into batches of up to f boxes that minimizes the walking distance covered by the pickers.

4.3 Homogeneous boxes

The boxes contain clothes that are exhibited in shops. When a box arrives at the shop, a salesperson carries it around the shop to prepare the displays. His/her job is made easier when boxes are *homogeneous*. Homogeneous means that a box contains clothes of a single class. Classes can be defined as needed, but an example of classes can be t-shirts, jeans, shirts, pulls, etc. For instance, a box containing 10 t-shirts is homogeneous, while a box containing one jeans and nine t-shirts is not.

Whenever homogeneity of the boxes is prior in the composition criteria, we propose to modify the split algorithm proposed in Section 4.1 to take this feature into account. The arc set defined by Equation (1) does not change. However, we modify the cost associated with each arc by increasing it by a factor β whenever it considers a non-homogeneous box. Thus, the cost of an arc (i,j) such that items $i+1, \ldots, j$ do not belong to the same class is $\alpha^{BOX} + \beta + d_{ij}$. When the value of β is increased, more homogeneous but also more numerous boxes are generated (especially when β and α^{BOX} are similar).

Moreover, if the minimal number of boxes required to contain the demand of a given customer, namely,

$$n_{box}^{min} = \max\left\{ \left\lceil \frac{\sum_{i \in \mathcal{I}} v_i}{v_{box}^{max}} \right\rceil, \left\lceil \frac{\sum_{i \in \mathcal{I}} w_i}{w_{box}^{max}} \right\rceil \right\}$$

is greater than a certain parameter γ, the order on which items are considered to build the acyclic graph is modified and does not depend anymore on the walking policy. In particular, items are first ranked such that items of the same class follow each other, and inside each class, items are ranked according to the walking policy. This increases the chance of having homogeneous boxes with the disadvantage of deteriorating the total walked distance.

5 Computational results

Our industrial partner provided us with a set of 487 instances. These instances come from 17 working days between January and November 2017. Working days are divided into 11 to 16 periods. For each

	Industrial solution	Proposed solution	Deviation in percentage (%)
Nb. of boxes	47 201	47 354	+0.32
Nb. of picking tours	8 105	8 103	-0.02
Nb. box violate volume min	575	33	-94.26
Total walking distance (m)	1 298 958	984 479	-24.21
Nb. homogeneous boxes	12 463	13 978	+12.16
Pct. of homogeneous boxes	26.40%	29.52%	-

Table 1: Global comparison with the industrial solutions.

period, one instance is created for the demands to be prepared in blocks 1 and 2, and another instance is created for the demands to be prepared in blocks 3 and 4. For each instance, the set of demands corresponds to a number of items to be picked that varies between 45 and 18 209, with an average of 2 469.

We run the algorithms presented in Sections 4.1–4.2 on the set of 487 instances to evaluate its efficiency. The parameters are set to keep a reasonable number of boxes and tours in the solution. Their value are: $\alpha^{BOX} = 30\ 000$ and $\alpha^{TOUR} = 15\ 000$. Each instance is solved on a Intel(R) Core(TM) i7-4600U CPU@2.10GHz, in less than one second. The overall results are reported in Table 1. From Table 1 we can notice that the number of boxes and the number of picking tours are similar in the solutions computed and in the solutions performed by HappyChic. However, the number of boxes with a volume lower than the minimal volume is drastically reduced in the proposed solution. Moreover, our approach drastically reduces the distance walked by pickers and, consequently increases their productivity. When we consider all the 487 instances individually, the proposed solution reduces the total distance in 462 instances. The deviation from the current industrial solution ranges from -90% to +8%, with an average of -21.85%. Hence the proposed solutions lead to significant improvements for the industrial partner.

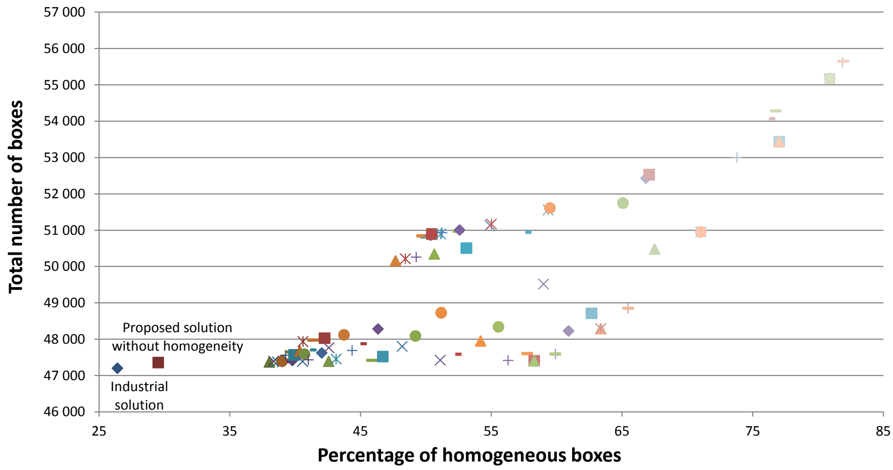
As explained in Section 4.3, the tasks of salespersons in shops are easier when boxes are *homogeneous*. In Figures 4a and 4b we present results obtained by the modified split algorithm proposed in Section 4.3 when homogeneity is explicitly taken into account. The blue dot in the left part of the figures corresponds to the industrial solution which is the worst among the others in term of homogeneity. The solution obtained with the algorithm presented in Section 4.1 without taken homogeneity into account is represented by the red square in the left part of the figures. This solution already improves the homogeneity of the boxes with respect to the industrial solution.

We have tested several values for parameters β and γ introduced in Section 4.3. β ranges from 5 000 to 40 000 with a step of 5 000, and γ ranges from 1 to 10 with a step of 1. Each combination of β and γ corresponds to a point in Figures 4a and 4b.

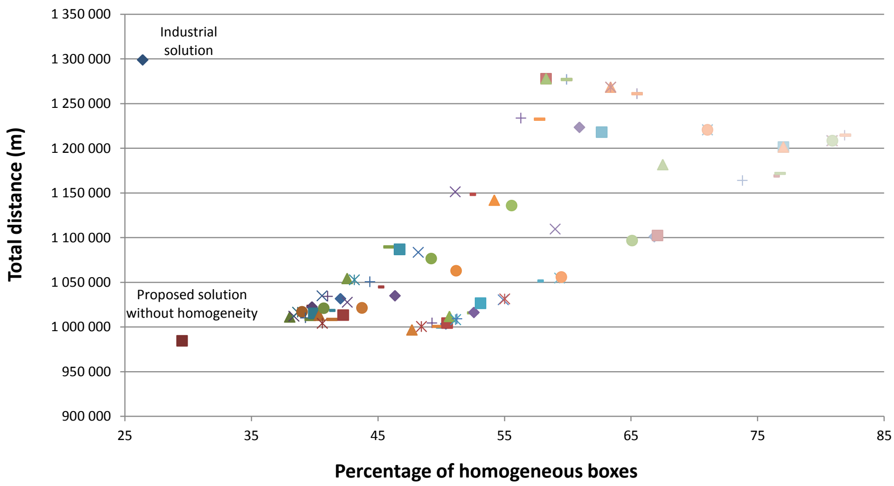
The homogeneity of the boxes can be improved by applying the changes described in Section 4.3 at the cost of degrading the total walking distance and the number of boxes. It is possible to reach 35% to 45% of homogeneous boxes with a small increase in the distance and the number of boxes. This is achieved with $\beta \leq 20\ 000$ and $\gamma \geq 6$. Homogeneity can reach up to 82%, at an expensive cost in the number of boxes. However, with 82% of homogeneous boxes, the distance is still lower than the distance covered in the current industrial solution.

6 Conclusions

In this paper, we presented with an integrated order batching and picker routing problem based on an industrial case encountered in a warehouse of the company HappyChic. The specific features of the studied problem are: (1) orders have to be split into several boxes, (2) boxes are batched into trolleys, and (3) the picking zone has a specific configuration, so the optimal routing policy is easy to determine when the locations to visit are known. To deal with large size instances, we proposed a heuristic algorithm based on a double split procedure: the first one to create boxes, the second one to batch



(a) Homogeneous boxes versus number of boxes.



(b) Homogeneous boxes versus total distance.

Figure 4: Solutions taking into account homogeneity: each point is a solution.

boxes into picking tours. The proposed algorithm can easily be adapted to take into account several classes of products with a homogeneity criterion when creating the boxes. The proposed results are very positive for the company: over a set of 487 instances, the total picking distance can be decreased by 24.21%.

References

Beasley, J. (1983). Route first–cluster second for vehicle routing methods, *Omega* **2**(4): 403–408.

de Koster, R., Le-Duc, T. and Roodbergen, K. (2007). Design and control of warehouse order picking: A literature review, *European Journal of Operational Reserach* **182**(2): 481–501.

Ecommerce Europe (2017). `https://www.ecommerce-europe.eu/`. [Online; accessed 24-October-2017].

Gademann, N. and van de Velde, S. (2005). Order batching to minimize total travel time in a parallel-aisle warehouse, *IIE Transactions* **37**(1): 63–75.

Marchet, G., Melacini, M. and Perotti, S. (2015). Investigating order picking system adoption: a case-study-based approach, *International Journal of Logistics: Research and Applications* **18**(1): 82–98.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem, *Computers & Operations Research* **31**(12): 1985–2002.

Scholz, A. and Wäscher, G. (2017). Order batching and picker routing in manual order picking systems: the benefits of integrated routing, *Central European Journal of Operations Research* **25**(2): 491–520.

Valle, C. A., Beasley, J. E. and da Cunha, A. S. (2017). Optimally solving the joint order batching and picker routing problem, *European Journal of Operational Research* **262**(3): 817 – 834.